

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problems Mailbox.**

(19)



JAPANESE PATENT OFFICE

PATENT ABSTRACTS OF JAPAN

(11) Publication number: **2000112708 A**

(43) Date of publication of application: **21 . 04 . 00**

(51) Int. Cl

G06F 3/14
G06F 12/00
G06F 13/00

(21) Application number: **11252241**

(22) Date of filing: **06 . 09 . 99**

(30) Priority: **25 . 09 . 98 US 98 161073**

(71) Applicant: **INTERNATL BUSINESS MACH CORP <IBM>**

(72) Inventor: **PI WEI CHIN
DANIEL GRAHAM DOUGLAS
EDWARD JOSEPH GARRAGER
BENJAMIN FRANKLIN E**

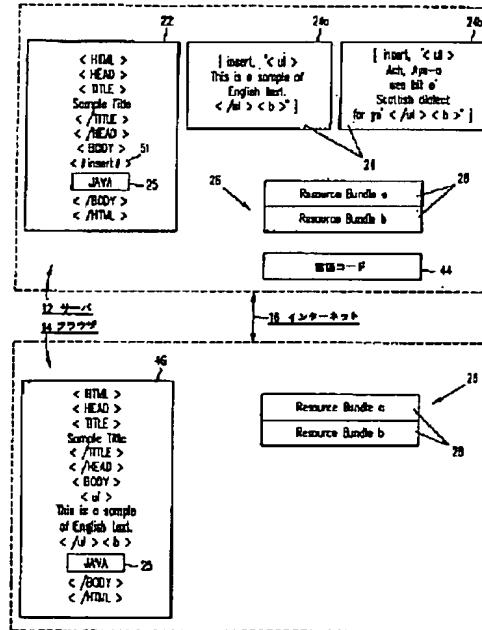
(54) USER INTERFACE AND ITS CONSTRUCTING METHOD

(57) Abstract:

PROBLEM TO BE SOLVED: To provide the user interface of a web base which is easily adaptable to various languages and a target viewer.

SOLUTION: A user interface constructing method is for the user interface (WUI) of the web base. A server 12 is provided with an HTML(hypertext markup language) template 22 and a JAR file 26 including an HTML Resource Bundle 24 and a Java Resource Bundle 24. In execution, a user selects a language by a browser 14 and a language code 44 is transmitted to a server 12. As the result, the server 12 constructs an HTML code 46 from the template 22 and the related bundle 24. The constructed code 46 (including a Java code 25) is transmitted to the browser 14 with the JAR file including a Java Resource Bundle 28.

COPYRIGHT: (C)2000,JPO



(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号
特開2000-112708
(P2000-112708A)

(43)公開日 平成12年4月21日 (2000.4.21)

(51)Int.Cl. ⁷	識別記号	F I	テーマコード ⁸ (参考)
G 0 6 F 3/14	3 1 0	G 0 6 F 3/14	3 1 0 C
12/00	5 4 6	12/00	5 4 6 R
13/00	3 5 4	13/00	3 5 4 D

審査請求 有 請求項の数14 O L (全 7 頁)

(21)出願番号 特願平11-252241
(22)出願日 平成11年9月6日(1999.9.6)
(31)優先権主張番号 09/161073
(32)優先日 平成10年9月25日(1998.9.25)
(33)優先権主張国 米国(US)

(71)出願人 390009531
インターナショナル・ビジネス・マシーンズ・コーポレーション
INTERNATIONAL BUSINESS MACHINES CORPORATION
アメリカ合衆国10504、ニューヨーク州
アーモンク(番地なし)
(72)発明者 ビ・ウェイ・チン
アメリカ合衆国95132、カリフォルニア州
サン・ホセ、カーマス・ドライブ 3139
(74)代理人 100086243
弁理士 坂口 博(外1名)

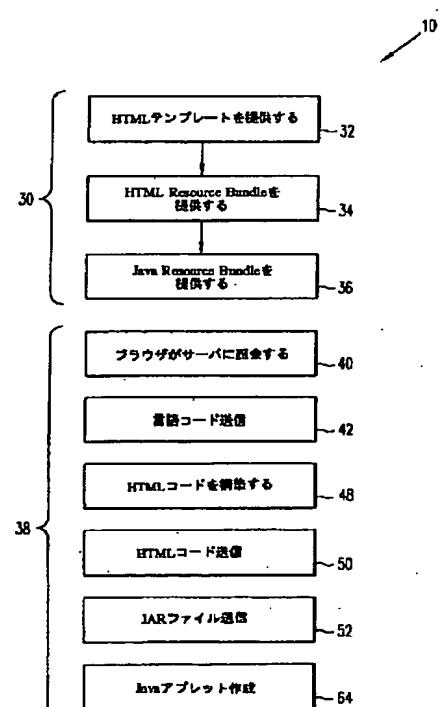
最終頁に続く

(54)【発明の名称】 ユーザ・インターフェース及びその構築方法

(57)【要約】

【課題】 さまざまな言語又は目標とする視聴者に容易に適応可能なウェブ・ベースのユーザ・インターフェースを提供する。

【解決手段】 ユーザ・インターフェース構築方法10はウェブ・ベースのユーザ・インターフェース(WUI)のためのものである。サーバ12は、HTMLテンプレート22、HTML Resource Bundle24、及びJava Resource Bundleを含むJARファイル26を持つ。実行時において、ユーザはブラウザ14にて言語を選択し、言語コード44はサーバ12へ送信される。その結果、サーバ12は、HTMLテンプレート22及び関連するHTML Resource Bundle24から、HTMLコード46を構築する。構築されたHTMLコード46(Javaコード25を含む)は、ブラウザ14へ、Java Resource Bundle28を含んでいるJARファイルとともに、送信される。



【特許請求の範囲】

【請求項1】置換変数をその中に持つHTMLコード化テンプレートと、前記置換変数を置き換えるためのデータを含む資源ファイルとを含む、ユーザ・インタフェース。

【請求項2】前記資源ファイルを複数含み、その選択された一の資源ファイルからのデータによって前記置換変数が選択的に置換えられる、請求項1に記載のユーザ・インタフェース。

【請求項3】複数の前記資源ファイルの夫々が、異なる言語のテキストを含む、請求項2に記載のユーザ・インタフェース。

【請求項4】サーバで前記HTMLコード化テンプレートと前記資源ファイルからのデータとを組み合わせることによりHTMLコードが作られる、請求項1に記載のユーザ・インタフェース。

【請求項5】複数の前記資源ファイルであって、構築されたHTMLコード・ページを作るために前記置換変数が選択された一の資源ファイルからのデータによって選択的に置換えられるようになっている複数の前記資源ファイルと、

前記HTMLテンプレート中のJavaコードと、Java Resource Bundleを含むJARファイルと、を更に含み、

前記構築されたHTMLコード・ページと前記JARファイルとがブラウザへ送信される、請求項1に記載のユーザ・インタフェース。

【請求項6】ウェブ・ベースのユーザ・インタフェースを構築するための方法であって、

少なくとも一つの変数を含むHTMLテンプレートをサーバへ提供するステップと、

夫々がその中に前記変数に対応するデータ部分を有する複数のデータ・ファイルをサーバへ提供するステップと、

前記複数のデータ・ファイルの一を選択するステップと、

前記複数のデータ・ファイルから選択された一のファイルからのデータ部分で前記HTMLテンプレートの中の前記変数を代えることによりHTMLコード化ユーザ・インタフェース・ファイルを構築するステップと、を含む、方法。

【請求項7】前記HTMLテンプレートがJavaコードを含み、

更に複数のJava Resource Bundleが提供され、前記Javaコードが実行される時に前記複数のJava Resource Bundleから選択された一のJava Resource Bundleからのデータがウェブ・ベースのユーザ・インタフェース中のJavaアプレットに提供される、請求項6に記載の方法。

【請求項8】前記複数のJava Resource Bundleは、JARファイルに結合され、前記HTMLコード化インタフェースと共にサーバからブラウザへ送信される、請求項7に記載の方法。

【請求項9】言語コードがブラウザからサーバへ送信され、

前記複数のデータ・ファイルの一が言語コードに従って選択される、請求項6に記載の方法。

【請求項10】複数のデータ・ファイルの夫々が、異なる言語のデータを含む、
請求項6に記載の方法。

【請求項11】変数をその中に有するHTMLテンプレートと、

複数のHTML Resource Bundleと、を含むコンピュータ・プログラムを記録したコンピュータ可読媒体。

【請求項12】前記HTML Resource Bundleの夫々が、前記変数に対する代替データを含む、
請求項11に記載のコンピュータ可読媒体。

【請求項13】選択されたデータでHTMLテンプレート中の変数を置き換えることによりHTMLコード化ユーザ・インタフェースをコンピュータに構築させるためのコードを含むコンピュータ・プログラムを記録したコンピュータ可読媒体。

【請求項14】コンピュータに前記HTMLコード化ユーザ・インタフェースをブラウザへ送信させるためのコードを更に含む、請求項13に記載のコンピュータ可読媒体。

【発明の詳細な説明】

【発明の属する技術分野】

【0001】本発明はコンピュータ・データ通信の分野に関し、特に特定の視聴者のために適応されたワールド・ワイド・ウェブ（以下、「WWW」と言う。）ユーザ・インタフェース・ページを提供する方法及びシステムに関する。本発明に係るウェブ・ベースのユーザ・インタフェース構築方法の適用範囲は、現在のところ、主としてWWWのためのウェブ・ページを提供することにある。そこでは、すべてが特定の視聴者に適応されている特定の言語、ダイアレクト、通貨等を提供する能力が求められている。

【0002】

【従来の技術】国際化は、プログラムを国際的に実行することを可能にするプロセスである。すなわち、国際化されたプログラムは、何処の国においても正しく実行するための柔軟性を、備えている。ひとたびプログラムが国際化されれば、特定の国及び言語のために単にローカライズ（地域化）するだけで、プログラムをその特定の言語で実行できるようになる。

【0003】国際化プロセスの一つのタスクは、プログラムのユーザ可視インタフェースをローカル言語へ変換

することである。しかしながら、求められているのはそれだけではない。他の問題は、日付及び時刻をその地域（ロケール）の慣習的な形式で表示すること、数字及び通貨をその地域の慣習的な形式で表示すること、並びにその地域の慣習的な順序で文字列をソートすること、を含む。

【0004】更にこれら全ての地域化の下にある問題は、より基本的な文字符号化の問題である。殆ど全ての有用なプログラムは、テキストの入力及び出力を実行できなければならない。そして、その結果ローカル文字標準に適応できることが必要である。

【0005】ウェブ・ベースのユーザ・インターフェース（以下、「WUI」と言う。）は、しばしばHTML及びJava（サン・マイクロシステムズ社の商標）の双方で実装される。各表示は、一又はそれ以上の要素（即ち、HTMLページ／フレームやJavaアプレット・パネル）からなる。WUIにおける変換は、これらの要素が異なる時に作られているという事実によって、複雑にされる（即ち、HTMLページ／フレームはダウンロードに先立ってウェブ・サーバ上にて実行されるアプリケーションによって作られており、一方でアプレット・パネルはエンド・ユーザのローカル・マシン上のメモリ中で実行されるJavaコードによって構築されている。）。

【0006】要するに、国際的市場への配布に適応可能とするために、ウェブ・ページはユーザへメッセージ（又は、GUIボタン上のラベルのような、他の何らかのユーザ可視テキスト）を現地域のための適切な言語及び形式で表示できるよう十分に柔軟でなければならぬ。これは典型的には、プログラムはハード・コードされたメッセージを使うべきではなくその代わりに実行時にローカル設定に基づくメッセージのセットを読み込まなければならないことを、意味する。Javaは、Javaアプレットのためにこれを行う途を提供する。Javaにおいて、メッセージは、キーと値の対として“ResourceBundle”サブクラス中に、定義される。そして、アプリケーションがサポートを意図する各言語又は地域のためにResourceBundleのサブクラスが作られ、ローカル・ネームを含む規則に従って各クラスをネーミングする。実行時において、ResourceBundle.getBundle()メソッドは、現地域のための適切なResourceBundleクラスをロードするために、用いられる。ResourceBundleは、アプリケーションが用いるメッセージを含む。その各メッセージは、メッセージ名として機能するキーに関連付けられる。この技術を用いて、アプリケーションは、地域独立メッセージ名に基づいて地域依存メッセージを検索することが出来る。

【0007】前述のように、Javaは、Javaアプレットの中で用いられる幾つかの解決策を提供してい

る。しかしながら、発明者の知識によれば、そのようなユーザ・インターフェースの殆どを構成するHTML（ハイパーテキスト・マークアップ言語）コード化インターフェース部のための対応する解決策は存在しない。現在までのところ、目標である視聴者に望まれているユーザ・インターフェースの各言語又は変化に対して異なったHTMLコードを提供することが、必要とされている。

【0008】

【発明が解決しようとする課題】それゆえに、さまざまな言語又は目標とする視聴者に容易に適応可能なWUIを提供する事が、本発明の一つの目的である。

【0009】受容可能な転送速度を達成するために限られた帯域幅しか必要としないWUIを提供することが、本発明の他の目的である。

【0010】多言語能力を獲得するためにブラウザ・ソフトウェアの変更を必要としないWUIを提供することが、本発明の更なる目的である。

【0011】特定の言語又は目標とする視聴者に適応された出力を提供するWUIを生成するための方法を提供することが、本発明の更なる目的である。

【0012】

【課題を解決するための手段】簡潔に言えば、本発明の好適な実施の形態は、Java言語に組み込まれた国際化能力を拡張し、データを記憶し及び通信するためのシステムである。そして更に、WUIの国際化を、キー変数及び関連する値を提供することにより、改善する。値は、サーバの中の複数のHTML ResourceBundleクラス（既知Java ResourceBundleクラスと同様な構造であるが、あとで詳述するように、内容は異なる）の中で提供され又は生成される。その際、ユーザ・インターフェースの通信が望まれる各言語又は他の同様な各変数に対して1つのHTML ResourceBundleが存在するようにされる。

【0013】ブラウザがユーザ・インターフェース・ページを要求し、所望の言語に関する情報をサーバへ提供すると、サーバは適切なHTML ResourceBundleからのキー及び値が埋め込まれたHTMLコードを用いて送信すべきHTMLコードを作る。可能性がある各言語のためのJava ResourceBundleを含んでいるJARファイルと共に、HTMLコード化インターフェース内のJavaアプレットもまた送信される。ブラウザにおいて、アプレットは、JARファイルからの適切なJava ResourceBundleを用いて、構築される。

【0014】本発明の利点は、WUIの国際化をより容易にすることである。

【0015】本発明の更なる利点は、国際化されたウェブ・ページがよりコンパクトにサーバに記憶されることである。

【0016】本発明のより更なる利点は、ブラウザへ転送されなければならないデータの総量が増加することなくウェブ・ページが国際化され得ることである。

【0017】

【発明の実施の形態】本発明を実現するための最も好適な実施の形態は、国際化のために必要とされている能力を含むユーザ・インターフェース構築方法である。本発明に係るユーザ・インターフェース構築方法は、図1のフローチャート中に図示されており、参照符号10によって示されている。

【0018】図2は、インターネット16を介して接続されているサーバ12及びブラウザ14の内容の関連部分を示しているブロック図である。そして、図3は、ブラウザ14によって表示される典型的なWU118のグラフィック表現である。本明細書において、サーバ12はコンピュータとその上で実行されるソフトウェアとの組み合わせを示し、ブラウザ14も同様にコンピュータとその上で実行されるソフトウェアとの組み合わせを示している。図3のWU118は、一般に図2のブラウザに関連付けられた表示スクリーン20上に表示される。

【0019】図2において、本発明により、サーバは、HTMLテンプレート22及び複数の（この例では簡単のためにただ2つのみ示されている）資源ファイル、すなわちHTML ResourceBundle24（この図中に符号24a及び24bとして示されている）を、持っている。以下でより詳細に議論されるように、"HTML ResourceBundle"という用語は、ここでは本発明に従った内容を持つ一又は複数のファイルとして定義される。ここに示されている例において、HTML ResourceBundle24のファイル構造は、標準的なJava ResourceBundleと異なっていない。そして、この例において、HTML ResourceBundle24と標準的なJava ResourceBundleとの間の差異は、内容にのみある。しかしながら、HTML ResourceBundle24中に含まれるものとしてここで議論されている情報を異なったファイル構造に含めることも可能である。

【0020】典型的に、HTMLテンプレート中には一又はそれ以上（この例ではただ一つのみ示されている）のJavaコード25のセグメントがある。図2が過度に複雑になることを避けるために、Javaコード25は、HTMLコード中に四角で囲まれた"JAVA"という語によって、表現されている。

【0021】JARファイル26は、Javaアプリケーションに関連する良く知られた型のファイル構造である。JARファイル26は、本発明によれば複数の（本例では二つ）のJava ResourceBundleを含む圧縮されたファイルである。

【0022】再び図1のフローチャートを図2との連係

において参照すると、本発明に係るユーザ・インターフェース構築方法10によれば、（サーバ12とブラウザ14との間の通信に先立って）一連の準備操作30中に、HTMLテンプレート22は、HTMLテンプレート提供操作32においてサーバ12に記憶される。同様に、HTML ResourceBundle24はHTML ResourceBundle提供操作34にてサーバ12に記憶され、JARファイル28はJava ResourceBundle提供操作36にてサーバ12に記憶（又は生成）される。

【0023】図1のフローチャート中に図示されている実行時操作38の議論に転ずると、ブラウザ14は、実行時にインターネット16を介して標準的な方法によってサーバ12に照会する。これは、図1のフローチャート中のブラウザによるサーバ照会操作40によって、示されている。言語コード送信操作42にて、言語コード44（図2参照）は、ブラウザ14からサーバ12へ送信される。幾つかのブラウザはサーバ12へ照会する時に言語コード44を自動的に送信する能力を持っているが、サーバ12がブラウザ14から更なる情報を要求できるようにする付加的ステップ（例えばブラウザ14におけるユーザ入力）が含まれ、それにより、サーバ12へ送られるべき言語コード44が生成されることとは本発明の範囲内である。例えば、言語以外に、ユーザ・インターフェース18中に表示される原文の長い又は短いバージョンのような代替要素がある場合、選択された言語コード44をブラウザ14からサーバ12へ送って、特定のHTML ResourceBundle24の選択によってそのような代替要素の選択を可能にすることは、本発明の範囲内である。しかしながら、一般には、どの言語が求められているかを示すための選択ボックスがサイン・オン・ページにある。以下では、ユーザがアメリカ英語形式のテキストを含むHTML ResourceBundle24aを選択するものとする。

【0024】実行時操作38について続けると、サーバ12は、HTMLコード構築操作48及びHTMLコード送信操作50で、選択されたHTML ResourceBundle24を用いて、HTMLコード46を構築し、それをブラウザ14へ送る。同様に、適切なJARファイル26が、JARファイル送信操作52でサーバ12からブラウザ14へ送信される。WU1の特定のページに対応するJava ResourceBundle28の全てを含んでいる完全なJARファイル26を送信するのが最も経済的であるが、言語コード44によって決定された、選択されたJava ResourceBundle28のみを送信することも本発明の範囲内である。

【0025】HTMLコード構築操作48で、HTMLテンプレート22中の置換変数51は、対応する選択されたHTML ResourceBundle24の内

容で、置換えられる。置換変数51は、HTMLコマンド区切り記号（“<”及び“>”）中にセットされ、ポンド記号（“#”）中に更にセットされることによって識別される。選択されたHTML Resource Bundle 24中で、対応する適切なテキストは、以下のように識別される。ResourceBundle 24中のデータ要素は、中括弧（“{”及び“}”）中にセットされることによって、分離され識別される。中括弧の中で、キー置換変数51は、HTMLテンプレート22中のそれと対応する。キー置換変数51に続いて、コンマ（“,”）があり、コンマの後に置換変数を置き換えるためのテキスト又は他の項目が続く。HTML ResourceBundleの置換内容はテキスト、HTML命令、画像ファイル、その他の、WUIを構築するためのHTML内容としてブラウザ14に認識される任意の内容を含み得る。

【0026】再び図3を参照すると、図示されている典型的なWUI18の例は、タイトル・フレーム54、ナビゲーション・フレーム56及びワーク・フレーム58を、持っている。何れかのフレーム、特にワーク・フレーム58中には、オプションとして一又はそれ以上（図3の例では一つが図示されている）のJavaアプレット60がある。Javaアプレットは、しばしば一又はそれ以上（図3の例では一つが図示されている）のボタン62又は当業者に知られた同様な機構を含む。Javaアプレット60は、JARファイル26からブラウザ14の選択言語によって決定された適切なJava ResourceBundle 28を用いて、ブラウザ14のマシンによって、実行時に構築される。この操作は、図1のフロー図中に、Javaアプレット作成操作64によって、示されている。慣習的な方法では、Javaアプレット60は、サーバ12とブラウザ14との間の通信が最小化されるように、ブラウザ14のユーザと動的に作用する。

【0027】図2に図示されたコード例が、本発明に従って、図3のワーク・フレームを作り出すことは、当業者であれば明らかであろう。

【0028】図示されているのは、ユーザ・インターフェースのシングル”ページ”の例である。本発明は、WUI18の複数の異なるページの利用可能性を提供するため、HTMLテンプレート22及びその関連要素の多*

*くの反復を含み得る。同様に、ここに記載された単純な例は二つのみのHTML ResourceBundle 24を持っているが、大多数のアプリケーションでは必要に応じてより多くの言語選択又は他の同様な提示を可能にするために、より多くのHTML ResourceBundle及び対応するJava ResourceBundle 28がある。

【0029】【産業上の適用性】本発明に係るユーザ・インターフェース構築方法10は、様々な言語又は他の任意の形式での表示に直ぐに適応可能なWUIを作るために広く使用されることを、意図する。

【0030】本発明に係る方法が普及すると、ウェブ・ページ構築エンジンは自動的にキーを挿入するため及び関連するHTML ResourceBundleを構築するために適応されることが、期待される。

【0031】前述のように、サーバ12及びブラウザ14の双方はインターネット16に接続されたコンピュータとして一般に実施され得る。HTMLテンプレート22、HTML ResourceBundle 24、及びJava ResourceBundle 26は、夫々実質的に任意のコンピュータ可読媒体中に実装され得る。そうすれば、夫々のデータをコンピュータ可読媒体からサーバ12のコンピュータへ転送することによって図1のフローチャートの操作32、34及び36を遂行することができる。同様に、図1のフローチャートとの関連で記載された操作は、コンピュータ可読媒体に同様に記憶されたコンピュータ可読プログラム・コード中に物理的に実施され、そこからサーバ12又は他の同様なコンピュータへ転送するようにしてもよい。

【0032】

【図面の簡単な説明】

【図1】本発明に係るWUI構築方法の例を示すフローチャート

【図2】インターネットを介して接続されたサーバ及びブラウザの内容の関係のある部分を示すブロック図

【図3】典型的なWUIのグラフィック表現を示す図

【符号の説明】

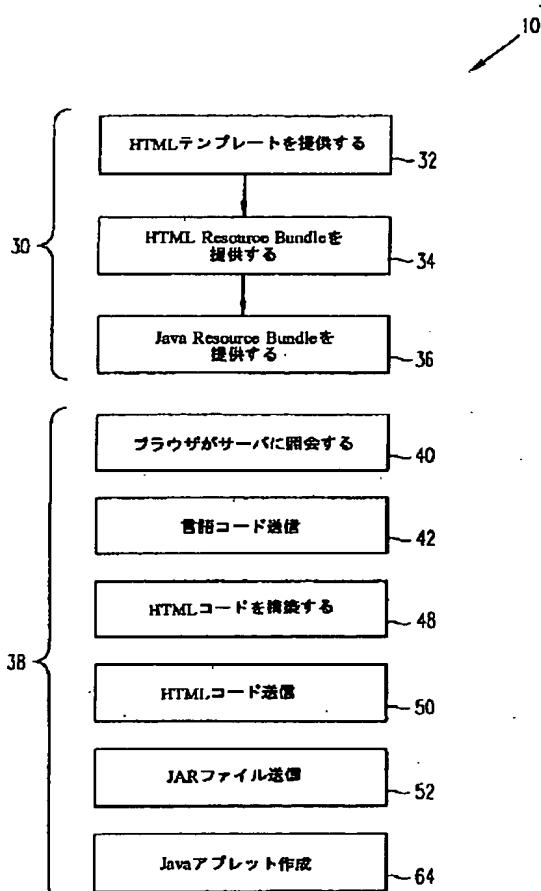
12 サーバ

14 ブラウザ

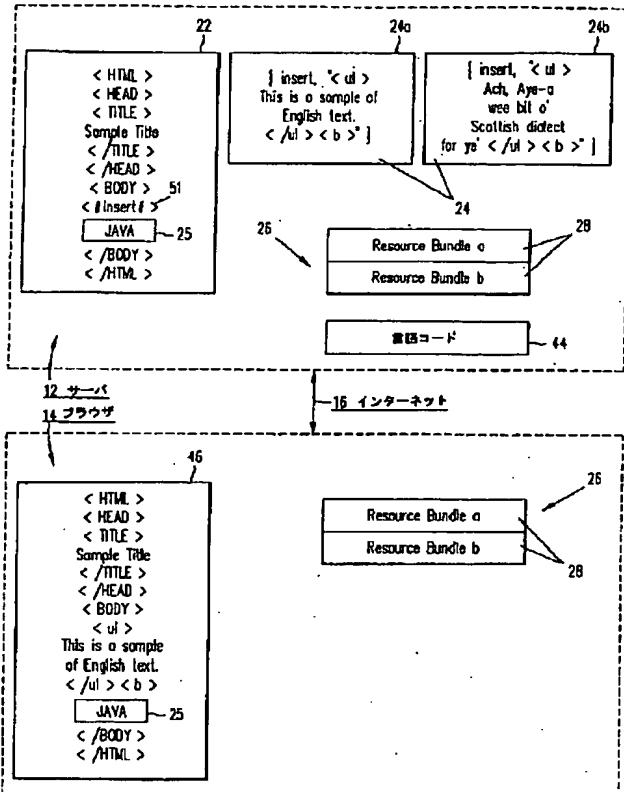
16 インターネット

18 WUI

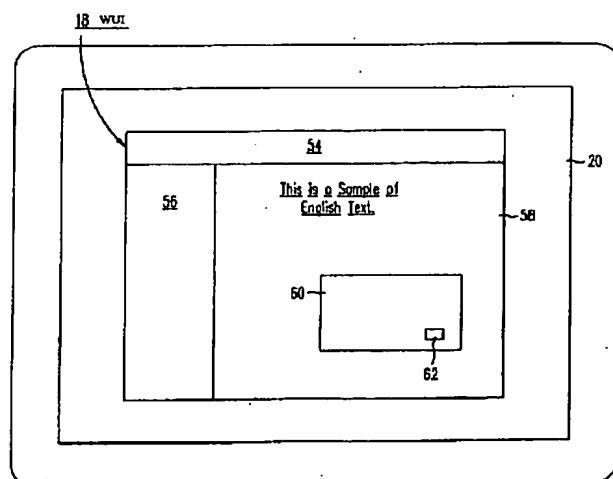
[図 1]



[図2]



(图3)



フロントページの続き

(72)発明者 ダニエル・グラハム・ダグラス
アメリカ合衆国95120、カリフォルニア州
サン・ホセ、プエルト・ヴァラルタ・ドライブ 1607

(72)発明者 エドワード・ジョセフ・ギャラガー
アメリカ合衆国95136、カリフォルニア州
サン・ホセ、キャピトル・ヴィレッジ・センター 542

(72)発明者 ベンジャミン・フランクリン・イー
アメリカ合衆国95120、カリフォルニア州
サン・ホセ、ブライア・ランチ・レーン 703